TP2	**Network reconstruction and analysis**
Janv.-Fév. 2019	**Durée: NA**

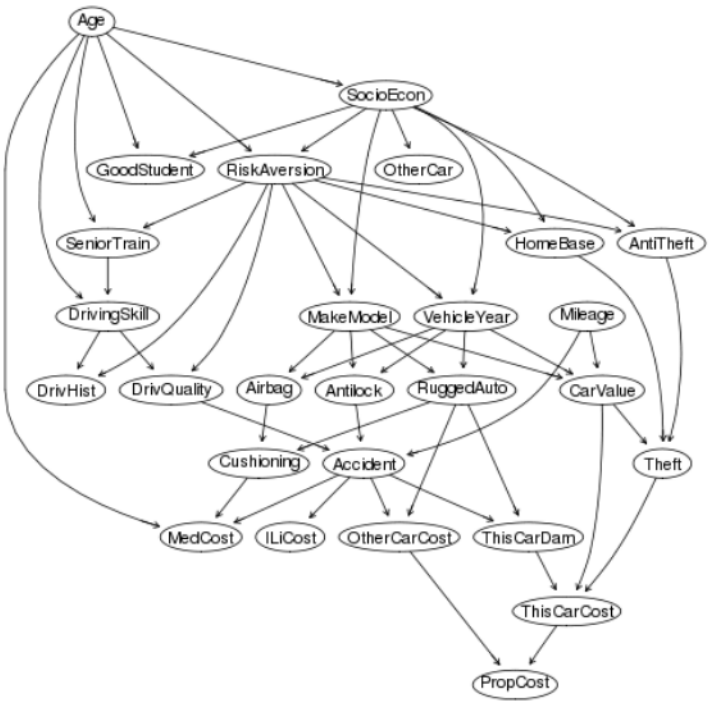
Objectif: Reconstruction et analyse d'un réseau - Evaluation

- Le TP peut être réalisé seul ou en binôme.

Network reconstruction - Evaluation

Problématique

We want to reconstruct a using various reconstruction methods. The objectives are to compare the different reconstructions and evaluate the inferred networks using the ground truth model. We will consider the *insurance* dataset proposed by the R package `bnlearn`, which is a network for evaluating car insurance risks (see R documents for details on variables). The R packages `igraph` and `pcalg` are also required for the following.



INSURANCE

Number of nodes: 27
Number of arcs: 52
Number of parameters: 984
Average Markov blanket size: 5.19
Average degree: 3.85
Maximum in-degree: 3

BIF (3.7kB)
DSC (3.1kB)
NET (2.4kB)
RDA (bn.fit) (3.7kB)
RDS (bn.fit) (3.7kB)

J. Binder, D. Koller, S. Russell, and K. Kanazawa. Adaptive Probabilistic Networks with Hidden Variables. Machine Learning, 29(2-3):213-244, 1997.

1. **Preliminaries**

   a. Install and load the following R packages: `bnlearn`, `igraph` and `pcalg`
   b. Create the *insurance* ground truth model from the model string (see *insurance* help)
   c. Check the class of the returned object and see the content.
   d. Get the adjacency matrix (`bnlearn::amat`).

e. Build a directed igraph network from the adjacency matrix and propose a (nice!) plot.

2. **Score-based method (*hill-climbing*)**

   a. Load the *insurance* data from the `bnlearn` package.

   b. Reconstruct the insurance network using the hill-climbing approach (`bnlearn::hc`). Check the class of the returned object and see the content.

   c. Get the adjacency matrix (`bnlearn::amat`).

   d. Build a directed igraph network from the adjacency matrix and propose a (nice!) plot.

   e. Count the number of true positive (TP), false positive (FP) and false negative (FN) (for the graph skeleton only). Compute *Precision*, *Recall* and *Fscore*.

   f. Highlight the FP edges in your reconstructed network.

   g. Propose a method to take the orientation into account.

3. **Constraint-based method (*PC*)**

   a. Reconstruct the insurance network using the PC approach (`pcalg::pc`) using the *disCItest* conditional independence test. You will need to perform the following transformations:

      – Convert your dataset to numeric using data.matrix
      – Make the categories start from 0.
      – Compute the number of levels for each variable
      – Prepare the *suffStat* object (see *pc* help)

   b. Get the adjacency matrix (`bnlearn::amat`).

   c. Build a directed igraph network from the adjacency matrix and propose a (nice!) plot.

   d. Count the number of true positive (TP), false positive (FP) and false negative (FN) (for the graph skeleton only). Compute *Precision*, *Recall* and *Fscore*.

   e. Highlight the FP edges in your reconstructed network.

4. **Local search method (*aracne*)**

   a. Reconstruct the insurance network using the PC approach (`bnlearn::aracne`).

   b. Get the adjacency matrix (`bnlearn::amat`).

   c. Build a directed igraph network from the adjacency matrix and propose a (nice!) plot.

   d. Count the number of true positive (TP), false positive (FP) and false negative (FN) (for the graph skeleton only). Compute *Precision*, *Recall* and *Fscore*.

   e. Highlight the FP edges in your reconstructed network.